

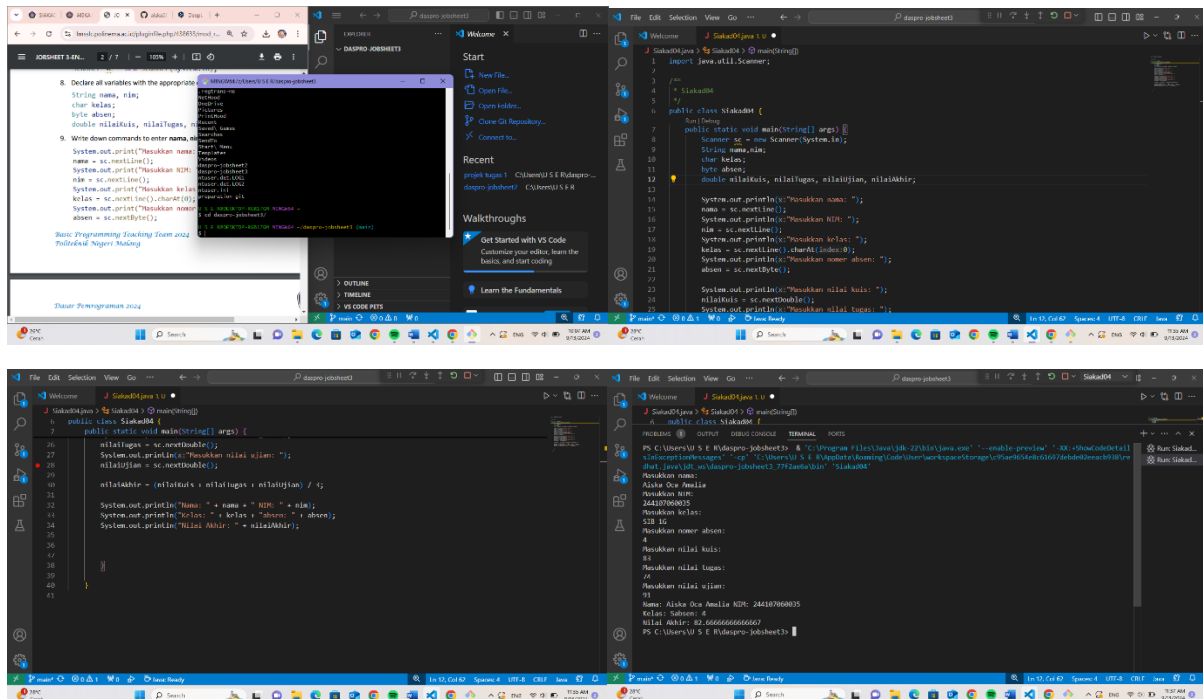
Name : Aiska Oca Amalia

NIM : 244107060035

Class : SIB 1G

JOBSHEET 3

2.1 Experiment 1: Case Study of Filling Student Grades at SIAKAD



2.1.3 Question

1. Why are the data type double used for nilaiKuis, nilaiTugas, and nilaiUjian? What happens if you use the int data type?

Answer: The double data type is used for decimal numbers. if we use the int data type the system we run can run but there will be missing numbers

2. Explain the meaning of the following program code!

Kelas = sc.nextLine().charAt(0);

Answer: This program code is useful for entering more than one character

3. Explain why the Scanner declaration is necessary?

Answer: scanner declaration is needed so that we can customase a program according to what we want to enter.

4. If the data in the class entered is equipped with the study program, for example **SIB 1G**, then what type of data should be used? Customize the program code!

Answer: we use the program code `charAt(0)`, in order to be able to enter more than one character.

5. Change the naming of the **nilaiAkhir** variable to **nilai-AkhirS**, compile and run the program! What are the results? Why is this the result?

Answer:

```

1 public class SiskadM4 {
2     public static void main(String[] args) {
3         Scanner scanner = new Scanner(System.in);
4         String nama;
5         long NPM;
6         String kelas;
7         double nilaiKuis, nilaiTugas, nilaiUTS, nilaiUAS;
8         double nilaiAkhir;
9
10        System.out.println("Masukkan nama: ");
11        nama = scanner.nextLine();
12        System.out.println("Masukkan NPM: ");
13        NPM = scanner.nextLong();
14        System.out.println("Masukkan kelas: ");
15        kelas = scanner.nextLine();
16        System.out.println("Masukkan nilai kuis: ");
17        nilaiKuis = scanner.nextDouble();
18        System.out.println("Masukkan nilai tugas: ");
19        nilaiTugas = scanner.nextDouble();
20        System.out.println("Masukkan nilai UTS: ");
21        nilaiUTS = scanner.nextDouble();
22        System.out.println("Masukkan nilai UAS: ");
23        nilaiUAS = scanner.nextDouble();
24
25        nilaiAkhir = (nilaiKuis * 0.2) + (nilaiTugas * 0.15) + (nilaiUTS * 0.3) + (nilaiUAS * 0.35);
26        System.out.println("Nilai Akhir: " + nilaiAkhir);
27    }
28 }
  
```

by adding S to the final value, there will be errors because it does not match the specified variable.

6. Modify the program code so that there are four component scores that are included to calculate the final score, namely quiz scores with a weight of 20%, assignment scores with a weight of 15%, UTS scores with a weight of 30%, and UAS scores with a weight of 35%!

Answer :

```

1 public class SiskadM4 {
2     public static void main(String[] args) {
3         Scanner scanner = new Scanner(System.in);
4         String nama;
5         long NPM;
6         String kelas;
7         double nilaiKuis, nilaiTugas, nilaiUTS, nilaiUAS;
8         double nilaiAkhir;
9
10        System.out.println("Masukkan nama: ");
11        nama = scanner.nextLine();
12        System.out.println("Masukkan NPM: ");
13        NPM = scanner.nextLong();
14        System.out.println("Masukkan kelas: ");
15        kelas = scanner.nextLine();
16        System.out.println("Masukkan nilai kuis: ");
17        nilaiKuis = scanner.nextDouble();
18        System.out.println("Masukkan nilai tugas: ");
19        nilaiTugas = scanner.nextDouble();
20        System.out.println("Masukkan nilai UTS: ");
21        nilaiUTS = scanner.nextDouble();
22        System.out.println("Masukkan nilai UAS: ");
23        nilaiUAS = scanner.nextDouble();
24
25        nilaiAkhir = (nilaiKuis * 0.2) + (nilaiTugas * 0.15) + (nilaiUTS * 0.3) + (nilaiUAS * 0.35);
26        System.out.println("Nilai Akhir: " + nilaiAkhir);
27    }
28 }
  
```

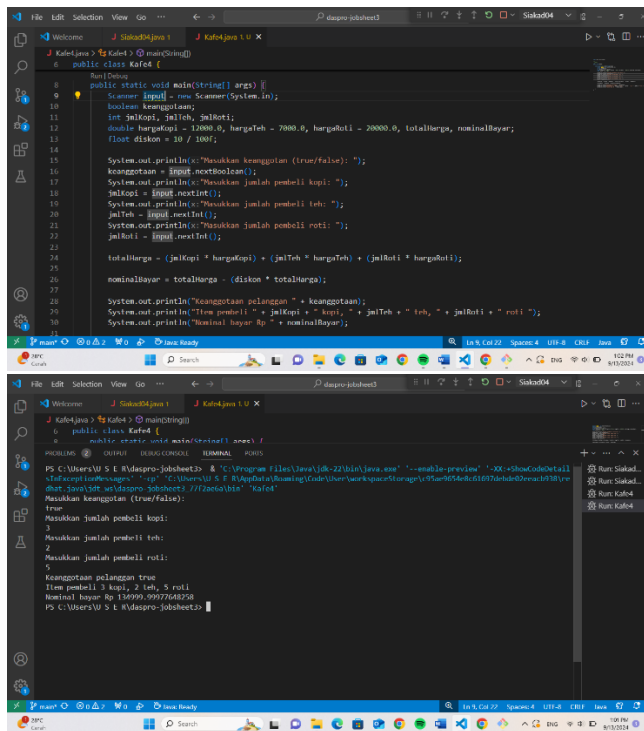
7. Commit and push program code to Github

Answer:

```

PS C:\Users\user> git add .
PS C:\Users\user> git commit -m 'Initial commit'
PS C:\Users\user> git push origin main
Enumerating objects: 3, done.
Counting objects: 100%, done.
Compressing objects: 100%, done.
Writing objects to disk: 100%, done.
Total 3 (delta 3), reused 0 (delta 0), pack reused 0 (from 0)
To https://github.com: [username]:main
 * [new branch]  main -> main
PS C:\Users\user>
  
```

2.2 Experiment 2: Case Study of Transactions in Cafes



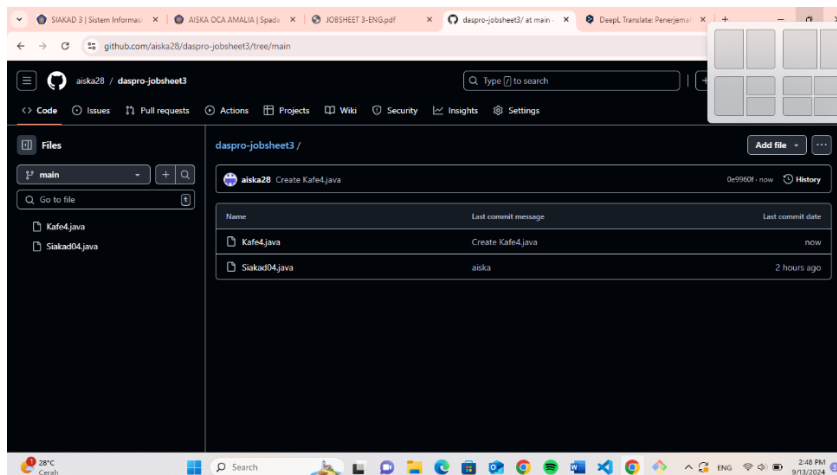
The top screenshot shows the Java source code for a cafe transaction simulation. The code defines a class `Kafe4` with a `main` method that prompts the user for the number of items (kopi, teh, roti), the discount status, and the item prices. It then calculates the total price and the final amount after applying the discount.

```
public class Kafe4 {  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
        boolean keanggotaan;  
        int jmlKopi, jmlTeh, jmlRoti;  
        double hargaKopi = 12000.0, hargaTeh = 7000.0, hargaRoti = 20000.0, totalHarga, nominalBayar;  
        float diskon = 10 / 100f;  
  
        System.out.println("Masukkan keanggotaan (true/false): ");  
        keanggotaan = input.nextBoolean();  
        System.out.println("Masukkan jumlah pembeli kopi: ");  
        jmlKopi = input.nextInt();  
        System.out.println("Masukkan jumlah pembeli teh: ");  
        jmlTeh = input.nextInt();  
        System.out.println("Masukkan jumlah pembeli roti: ");  
        jmlRoti = input.nextInt();  
  
        totalHarga = (jmlKopi * hargaKopi) + (jmlTeh * hargaTeh) + (jmlRoti * hargaRoti);  
        nominalBayar = totalHarga - (diskon * totalHarga);  
  
        System.out.println("Keanggotaan pelanggan = " + keanggotaan);  
        System.out.println("Item pembeli = " + jmlKopi + " kopi, " + jmlTeh + " teh, " + jmlRoti + " roti.");  
        System.out.println("Nominal bayar Rp = " + nominalBayar);  
    }  
}
```

The bottom screenshot shows the execution output of the program. It prompts the user for the number of items and the discount status, and then displays the calculated total price and the final amount after applying the discount.

```
PS C:\Users\S E R\Videos\Jobsheet3> java -cp ".\bin" Kafe4  
Masukkan keanggotaan (true/false):  
true  
Masukkan jumlah pembeli kopi:  
3  
Masukkan jumlah pembeli teh:  
2  
Masukkan jumlah pembeli roti:  
5  
Keanggotaan pelanggan true  
Item pembeli = 3 kopi, 2 teh, 5 roti  
Nominal bayar Rp 134999.99977648258  
PS C:\Users\S E R\Videos\Jobsheet3>
```

11. Commit and push program code to Github



2.1.3 Question

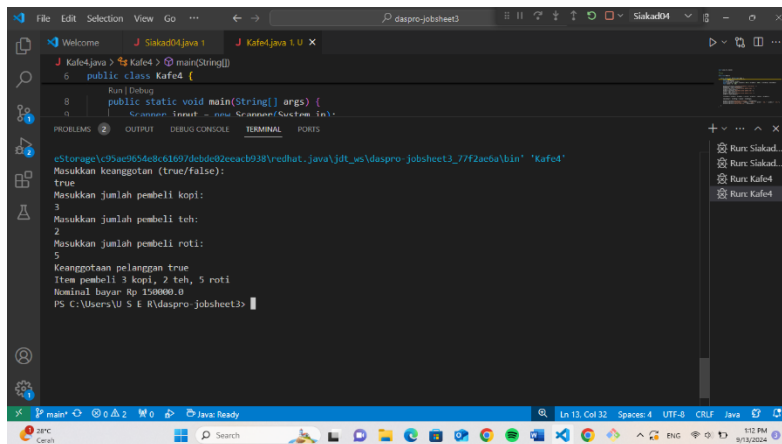
1. What does the addition of the letter 'f' mean in the initialization of the following variable?

Float diskon = 10 / 100f;

Answer: the letter 'F' is useful for automatically processing discounts

2. What happens if the letter 'f' in question number 1 is removed? Compile and run, then compare the results before and after the removal of the 'f'!

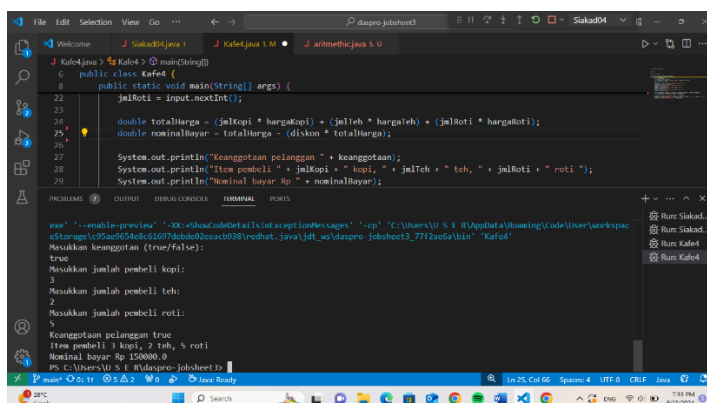
Answer:



if the letter 'f' is omitted then the discounting process will not occur automatically.

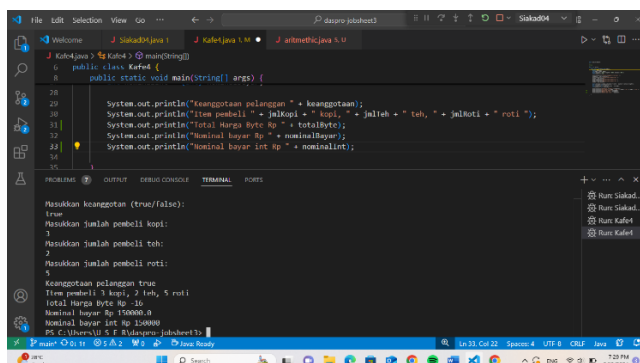
3. Add the variable `nominalInt` after the calculation of `nominalBayar` to accommodate the nominal pay with type `int`, then cast from `double` to `int`, and display the result!

Answer:



4. Add the totalByte variable after the calculation of the totalHarga to hold the total price by byte type, then cast from double to byte, and display the result!

Answer:



5. In question number 4, why is the result so?

Answer: by using the code it will experience doubling

6. What is the function of casting? Why is casting necessary?

Answer: serves to change the data type of a primitive data or an object

3. Assignment

1. An electricity customer wants to know his total electricity bill. Electricity tariffs are calculated based on the amount of electricity used in kilowatt-hours (kWh). The electricity tariff is Rp 1,500 per kWh. There is a check on electricity usage whether it exceeds 500 kWh (using a relationship operator with a boolean type). Identify the inputs, outputs, and algorithms, then code the program!

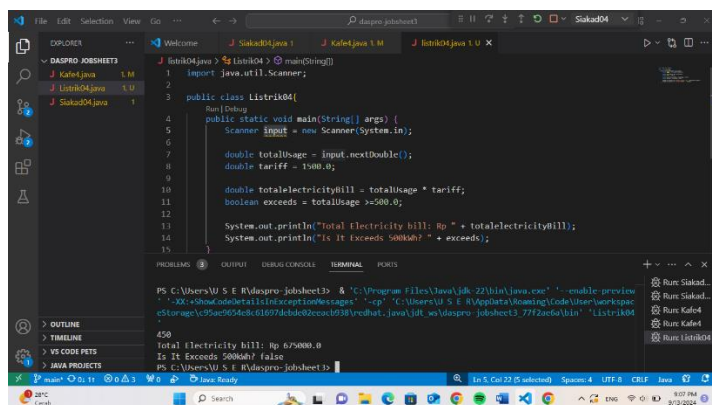
Answer:

Input : totalusage,tariff

Output: total bill, exceeds

Algoritma: Start multiplying the total usage by the previous tariff, by comparing greater or less than 500 kWh, enter the value. program successful

Code program:

The screenshot shows an IDE with a Java file named 'Listrik04.java'. The code defines a class 'Listrik04' with a 'main' method. It uses a 'Scanner' to take input for 'totalUsage' and 'tariff'. It calculates 'totalElectricityBill' by multiplying 'totalUsage' by 'tariff'. It then checks if 'totalUsage' is greater than or equal to 500.0 and prints the result. The output window shows the program running successfully with the following output: 'Total Electricity bill: Rp 675000.0' and 'Is it Exceeds 500kwh? false'.

```
1 import java.util.Scanner;
2
3 public class Listrik04 {
4     public static void main(String[] args) {
5         Scanner input = new Scanner(System.in);
6
7         double totalUsage = input.nextDouble();
8         double tariff = 1500.0;
9
10        double totalElectricityBill = totalUsage * tariff;
11        boolean exceeds = totalUsage >= 500.0;
12
13        System.out.println("Total Electricity bill: Rp " + totalElectricityBill);
14        System.out.println("Is it Exceeds 500kwh? " + exceeds);
15    }
16 }
```

2. A company wants to create a simple program to calculate the monthly salary of its employees. Employee salaries are calculated based on the number of hours worked and hourly wages. In addition, employees also get a bonus of 10% of the total salary before tax. After that, a tax of 5% is imposed on the salary and bonus that has been calculated. Identify the inputs, outputs, and algorithms, then code the program!

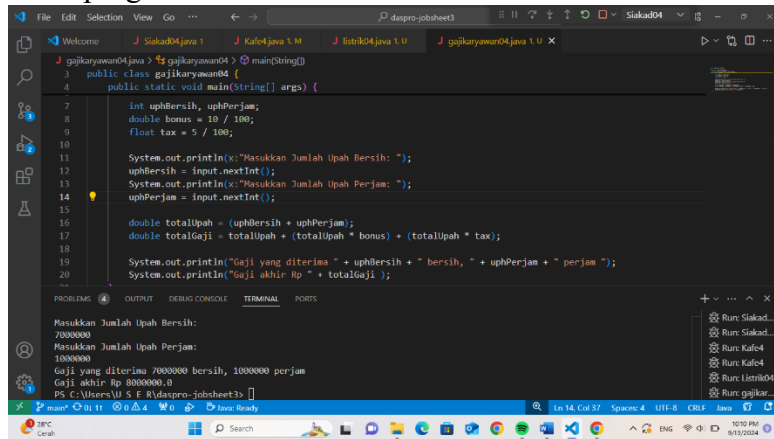
Answer:

Input: uphBersih, uphPerjam, bonus, tag

Output: total salary, total net salary

Algoritma: start calculating the sum of net wage and hourly wage, then multiply by the predetermined bonus percentage and tex percentage, execute and input the values. finish stop

Code program:



The screenshot shows an IDE with a Java file named `gajikaryawan04.java`. The code defines a `main` method that prompts the user for 'Masukkan Jumlah Upah Bersih' and 'Masukkan Jumlah Upah Perjam'. It then calculates the total salary by adding the clean wage, a 10% bonus, and a 5% tax. The output shows a clean wage of 7000000, a per-hour wage of 1000000, and a final salary of 8800000.0.

```
1 public class gajikaryawan04 {
2     public static void main(String[] args) {
3
4         int uphBersih, uphPerjam;
5         double bonus = 10 / 100;
6         float tax = 5 / 100;
7
8         System.out.println("Masukkan Jumlah Upah Bersih: ");
9         uphBersih = input.nextInt();
10        System.out.println("Masukkan Jumlah Upah Perjam: ");
11        uphPerjam = input.nextInt();
12
13        double totalUpah = (uphBersih + uphPerjam);
14        double totalGaji = totalUpah * (totalUpah * bonus) + (totalUpah * tax);
15
16        System.out.println("Gaji yang diterima " + uphBersih + " bersih, " + uphPerjam + " perjam ");
17        System.out.println("Gaji akhir Rp " + totalGaji);
18    }
19 }
20
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan Jumlah Upah Bersih:
7000000
Masukkan Jumlah Upah Perjam:
1000000
Gaji yang diterima 7000000 bersih, 1000000 perjam
Gaji akhir Rp 8800000.0
PS C:\Users\U S E R\daspro-jobsheet3>

Run: Slakad...
Run: Slakad...
Run: Kafe4
Run: Kafe4
Run: Istrik04
Run: gajikar...

main 0.11 0.04 Java Ready

Ln 14, Col 37 Space: 4 UTF-8 CRLF Java

28°C
Lelah